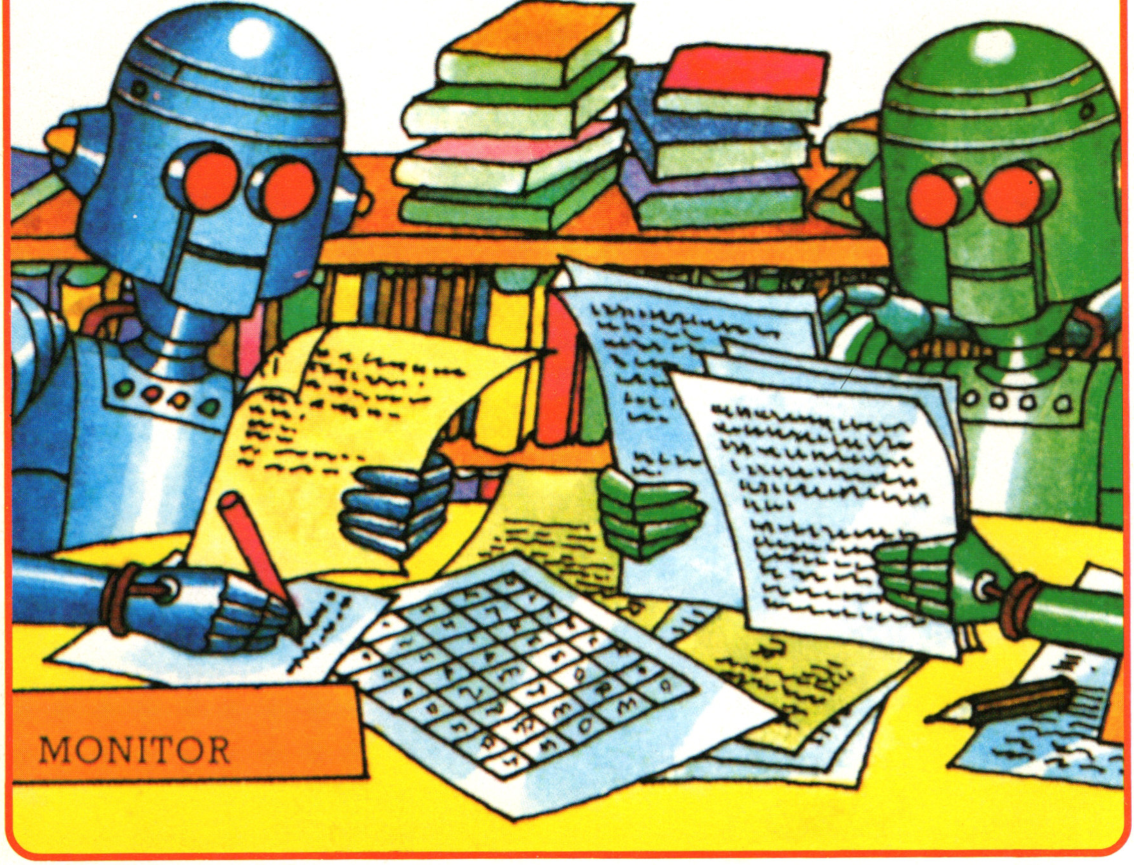
L. 2.500
Frs. 3,75

BEST-SELLER MONDIALE

LA GRANDE ENCICLOPEDIA DI INFORMATICA PER RAGAZZI

29

IN SOLI 30 FASCICOLI



GRUPPO EDITORIALE JACKSON

IN COLLABORAZIONE CON AMSTRAD

Spedizione in Abb. Postale Gruppo II/70

Direttore responsabile
Paolo Reina

Direttore di divisione:
Roberto Pancaldi

Autori:
Judy Tatchell,
Nick Cutler,

Lisa Watts,
Mike Wharton,

Tony Potter,
Ivor Guild,

Ian Graham,
Lynn Myring,

Helen Davies,
Mike Wharton,

Ian Graham,

Brian Reffin Smith,
Lisa Watts,

Bill Bennett,
Judy Tatchell,

Jenny Tyler,

Lee Howarth,
Judy Tatchell,

Gaby Waters,
Graham Round,

Nick Cutler,
Gaby Waters,

Brian Reffin Smith,

Judy Tatchell,
Lee Howarth,

Cherry Evans,
Lee Howarth

Revisione e adattamento:
Martino Sangiorgio

Coordinamento editoriale:
Renata Rossi

Progetto grafico:
Sergio Mazzali

Distribuzione:
SODIP - Milano

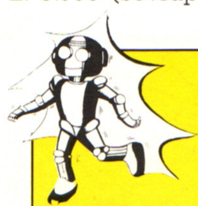
Stampa:
Vela - WEB - Vigano di Gaggiano (MI)

Direzione e Redazione:
Via Rosellini, 12 - Milano (20124) - Tel. 02/6880951 (5 linee)

Autorizzazione alla pubblicazione: Tribunale di Milano n° 226 del 28/3/89.
Spedizione in abbonamento postale Gruppo II/70
(autorizzazione della Direzione Provinciale delle PPTT di Milano)
Prezzo del fascicolo L. 2.500
I numeri arretrati saranno disponibili per 1 anno dal completamento dell'opera e potranno essere richiesti direttamente all'Editore a L. 3.000 (sovrapprezzo di L. 10.000 per spese d'imballo e spedizione).
I versamenti vanno indirizzati a:

Gruppo Editoriale Jackson S.p.A.
Via Rosellini, 12 - 20124 Milano

mediante emissione di assegno bancario oppure utilizzando il C.C. Postale N. 11666203.
Non vengono effettuate spedizioni in contrassegno.



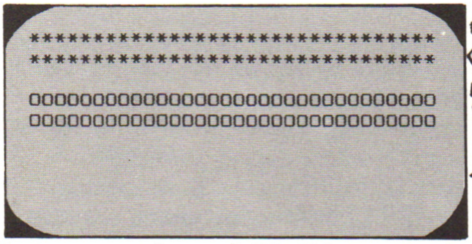
NEL PROSSIMO NUMERO:

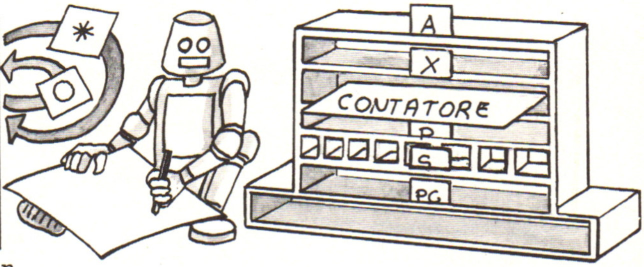
- **MNEMONICI E CODICI ESADECIMALI PER IL 6502**
- **LESSICO DEL LINGUAGGIO MACCHINA**
- **GLOSSARIO PER PERSONAL COMPUTER**
- **LISTATO PROGRAMMA "MENU" PER SOTTERRANEO DEL FATO**
- **DISEGNARE CON IL TURTLE**
- **COME INSEGNARE AI ROBOT**
- **TITOLATRICI ELETTRONICHE**
- **I CD-ROM**
- **INDICE SECONDO VOLUME**

Lampeggiamento dello schermo sul 6502

Questo programma scambia i due blocchi, byte dopo byte (cioè carattere dopo carattere), partendo con l'ultimo byte di ciascun blocco. Carica questi byte nei registri, poi deposita il byte del blocco a nella locazione di schermo del blocco b e viceversa. Poi il programma è ripetuto per scambiare la coppia seguente di byte.

a
b





Il programma fa uso dell'indirizzamento con indice per trovare gli indirizzi di ciascun byte. Il numero complessivo di byte in un blocco è caricato nel registro X; poi, per depositare o caricare un byte, il numero nel registro X è sommato all'indirizzo di inizio di ciascun blocco. L'istruzione DEX (decrementa X) fa sottrarre al computer 1 da X, cosicché, quando il programma si ripete, il computer deposita il byte seguente di nuovo sullo schermo.

Programma 6502

Cercate a pagina 436 come ricavare i valori di n, a e b. Poi sottraete 1 da a e da b, cosicchè quando il computer somma loro X ottiene l'ultimo indirizzo di ciascun blocco, piuttosto che il primo indirizzo della linea seguente. (Assicuratevi che n, a e b siano in esadecimale).

Mnemonici	Cod.esad.	Significato
LDX # n	A2 n	Carica X con il numero di byte di un blocco.
LDA indirizzo a,X	BD indirizzo a	Metti il contentuto della locazione con l'ind. a+X nell'accumul.
TAY	A8	Trasferisci il contenuto dell'accumulatore nel registro Y.
LDA indirizzo b,X	BD indirizzo b	Metti il contenuto della locazione con l'ind. b+X nell'accumulat.
STA indirizzo a,X	9D indirizzo a	Deposita il contenuto dell'accumulatore all'indirizzo a+X.
TYA	98	Trasferisci il contenuto del registro Y di nuovo all'accumulatore.
STA indirizzo b,X	9D indirizzo b	Deposita il contenuto dell'accumulatore all'indirizzo b+X.
DEX	CA	Decrementa X. Lo zero flag è posto a 1 quando X=0.
BNE all'istruzione due	D0 EF	Dirama indietro alla locazione &EF se X è diverso da 0. EF è l'esadecimale per il complemento a due di 17 (vedi pag. 435)
RTS	60	RETURN.

Caricare ed eseguire il programma per lo Z80 o per il 6502

La migliore maniera di eseguire questo programma è come subroutine in linguaggio macchina nell'hex loader. Per fare questo, seguite questi passaggi:

1. Inserite l'hex loader e mettete i codici esadecimali per il microprocessore del vostro computer alla linea 160.

2. Alla linea 180 avete bisogno di due loop per inserire i caratteri nella memoria video. Per esempio, ecco le linee per due righe di * (codice 42) seguite da due righe di 0 (codice 48), per un computer con schermo di 40 colonne.

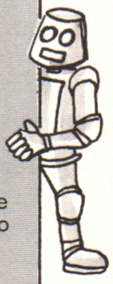
```
180 FOR J=0 TO 79
190 POKE primo indirizzo dello
    schermo + J, 42
200 NEXT J
210 FOR J=80 TO 159
220 POKE primo indirizzo dello
    schermo + J, 48
230 NEXT J
```

3. In seguito aggiungete le seguenti linee alla fine del programma:

```
240 CALL indirizzo a cui e' de-
    positato il codice macchina
250 FOR K=1 TO 500
260 NEXT K
270 GOTO 240
```

Adattate il numero 500 nel loop di ritardo per adattarsi al vostro computer.

4. Ora battete RUN per eseguire il programma. L'hex loader inserisce i codici esadecimali nella memoria, poi inserisce i codici dei caratteri nella memoria video. La linea 240 lo fa andare alla locazione in cui è depositato il codice macchina ed eseguire le istruzioni. Da solo il programma in linguaggio macchina scambia i blocchi solo una volta, cosí la linea 270 gli fa richiamare il programma volta dopo volta per generare l'effetto di lampeggiamento. Avete bisogno di un loop di ritardo perché il linguaggio macchina è molto veloce.



Andare oltre

Se volete sapere di piú sul linguaggio macchina il modo migliore è tentare di scrivere brevi programmi e provare a studiare programmi scritti da altri. Provate ad usare il linguaggio macchina nella veste di breve subroutine per eseguire un compito particolare all'interno di un programma in BASIC. Per esempio il linguaggio macchina è particolarmente adatto ad ordinare dati o a riempire lo schermo di grafici, poiché è piú veloce ed occupa meno spazio in memoria del BASIC. Potete trovare delle subroutine per fare cose di questo genere nelle riviste. Se le subroutine sono scritte per il vostro computer potete eseguirle senza alterazioni; se invece sono scritte per un'altra marca di computer che usa lo stesso microprocessore avrete bisogno di cambiare qualsiasi indirizzo del programma con gli indirizzi dell'area della memoria del vostro computer che avete scelto per depositarvi il codice macchina.

Le subroutine in linguaggio macchina

Ecco qui i passi che dovete seguire per usare una subroutine in linguaggio macchina all'interno di un programma BASIC.

1. Fate spazio nella memoria per il codice macchina, abbassando la cima della RAM utente.

2. Mettete i codici della subroutine in linguaggio macchina nella linea 160 del programma hex loader.
(Assicuratevi che ci sia un'istruzione di ritorno alla fine del programma in linguaggio macchina). Aggiungete delle linee per inserire qualsiasi byte di dati, se necessario, poi scrivete ed eseguite l'hex loader.

3. Numerate il vostro programma in BASIC usando i numeri di linea dopo quelli usati nell'hex loader. Nel punto in cui volete che il computer esegua il codice macchina, mettete, come una linea del programma BASIC, il comando adatto al vostro computer per eseguire un programma in linguaggio macchina.



4. Scrivete il programma in BASIC nel vostro computer, poi battete RUN. Il computer eseguirà le istruzioni BASIC e quando raggiungerà la linea che gli dice di eseguire il programma in linguaggio macchina andrà all'indirizzo in cui è depositato il codice macchina ed eseguirà le istruzioni. L'istruzione di ritorno alla fine del programma in linguaggio macchina manderà alla linea successiva del programma BASIC.

Usare un assembler

Un assembler (un programma che vi consente di scrivere un programma in linguaggio macchina con gli mnemonici) rende la programmazione in linguaggio macchina di gran lunga più facile. Potete acquistare un assembler su cassetta per la maggior parte degli home computer e alcuni hanno un assembler compreso nella macchina. Con un assembler potete scrivere commenti a lato degli mnemonici per ricordarvi che cosa fa ciascuna linea. L'assembler mostrerà poi sullo schermo il programma in mnemonici ed esadecimali, con gli indirizzi in cui sono depositate le istruzioni ed i commenti. L'assembler invertirà automaticamente le coppie di cifre degli indirizzi e ricaverà gli indirizzi o gli spostamenti per un salto. Alcuni assembler vi consentono di usare nomi simbolici per i dati, come le variabili in BASIC. Un buon assembler ha anche un "debugger" per trovare errori ed un "editor" per aiutarvi a correggerli.

Libri suggeriti

Ci sono parecchi libri sul linguaggio macchina, scritti appositamente per un particolare modello di microcomputer. La maniera migliore per sceglierne uno è di leggere le recensioni nelle riviste di computer. Potete inoltre trovare utili i seguenti libri:
Programmazione dello Z80 e Programmazione del 6502, entrambi di Rodnay Zaks e pubblicati in Italia dal Gruppo Editoriale Jackson. Sono guide molto dettagliate con liste complete di tutte le istruzioni di ciascun microprocessore. Non sono semplici da leggere per dei principianti, ma sono utili per la consultazione.
Anche per i processori più recenti (8086/ 8088, 68000, 80286, 80386), si possono trovare libri adeguati presso il Gruppo Editoriale Jackson.

Mnemonici e codici esadecimali per lo Z80

Gli mnemonici ed i codici esadecimali delle istruzioni coperte in questo libro sono forniti sulle pagine che seguono. Il termine "indirizzamento implicito" usato in queste liste è solo il nome di istruzioni per cui non è necessario specificare alcun operando nel codice esadecimale. Ci sono alcune altre istruzioni non elencate qui e se volete andare oltre con il linguaggio macchina avrete bisogno di una lista completa del set di istruzioni dello Z80 (vedi pagina precedente). Le seguenti abbreviazioni sono usate in queste liste:

n = numero
nn = numero di due byte
r = registro
rr = coppia di registri
x = indirizzo
c = condizione
d = spostamento

ADC A,n Somma con riporto un numero, n, all'accumulatore, (Indirizzamento immediato)

ADC A,n	CE,n

ADC A,r Somma con riporto il registro r all'accumulatore, (Indirizzamento implicito).

ADC A,A	8F
ADC A,B	88
ADC A,C	89
ADC A,D	8A
ADC A,E	8B
ADC A,H	8C
ADC A,L	8D

ADC HL,rr Somma con riporto il contenuto della coppia di registri rr con HL. (Indirizzamento implicito).

ADC HL,BC	ED4A
ADC HL,DE	ED5A
ADC HL,HL	ED6A

ADD A,n Somma con numero, n, all'accumulatore. (Indirizzamento immediato).

ADD,n	C6,n

ADD A,r Somma il registro r all'accumulatore. (Indirizzamento implicito).

ADD A,A	87
ADD A,B	80
ADD A,C	81
ADD A,D	82
ADD A,E	83
ADD A,H	84
ADD A,L	85

ADD HL,rr Somma il contenuto della coppia di registri rr ad HL. (Indirizzamento implicito).

ADD HL,BC	09
ADD HL,DE	19
ADD HL,HL	29

CALL x Va alla subroutine che inizia all'indirizzo x. (Indirizzamento immediato).

CALL x	CD x

CALL c,x Va alla subroutine che inizia all'indirizzo x, se si verifica la condizione c. c può essere Z (uguale); NZ (non uguale); C (riporto); NC (nessun riporto); PE (parità pari); PO (parità dispari); M (negativo) o P (positivo). (Indirizzamento immediato).

CALL Z,x	CC,x
CALL NZ,x	C4,x
CALL C,x	DC,x
CALL NC,x	D4,x
CALL PE,x	EC,x
CALL PO,x	E4,x
CALL M,x	FC,x
CALL P,x	F4,x

CCF Complementa il carry flag. (indirizzamento implicito).

CCF	3F

CP n Confronta il contenuto dell'accumulatore con il dato n. (indirizzamento immediato).

CP n	FE n

CP r Confronta il contenuto del registro r con l'accumulatore. (indirizzamento implicito).

CP A	BF
CP B	B8
CP C	B9
CP D	BA
CP E	BB
CP H	BC
CP L	BD

CP(HL) Confronta il contenuto dell'accumulatore con il contenuto dell'indirizzo posto in HL. (indirizzamento indiretto).

CP (HL)	BE

DEC r Decrementa il registro r. (indirizzamento implicito).

DEC A	3D
DEC B	05
DEC C	0D
DEC D	15
DEC E	1D
DEC H	25
DEC L	2D

DEC rr Decrementa la coppia di registri rr. (indir. implicito).

DEC BC	0B
DEC DE	1B
DEC HL	2B
DEC IX	DD2B
DEC IY	FD2B

DEC(HL) Decrementa il contenuto dell'indirizzo posto in HL.(indirizzamento indiretto).

DEC (HL)	35

INC r Incrementa il registro r. (indirizzamento implicito).

INC A	3C
INC B	04
INC C	0C
INC D	14
INC E	1C
INC H	24
INC L	2C

INC rr Incrementa la coppia di registri rr. (indiriz. implicito).

INC BC	03
INC DE	13
INC HL	23

INC(HL) Incrementa il contenuto dell'indirizzo posto in HL. (indirizzamento indiretto).

INC (HL)	34

JP x Salta all'indirizzo x. (indirizzamento immediato).

JP x	C3 x

JP(rr) Salta all'indirizzo posto nella coppia di registri rr. (indirizzamento implicito).

JP (HL)	E9
JP (IX)	DDE9
JP (IY)	FDE9

JP c,x Salta all'indirizzo x se si verifica la condizione c. c può essere Z (uguale); NZ (non uguale); C (riporto); NC (nessun riporto); PE (parità pari); PO (parità dispari); M (negativo) o P (positivo). (indirizzamento immediato).

JP Z,x	CA,x
JP NZ,x	C2,x
JP C,x	DA,x
JP NC.x	D2,x
JP PE,x	EA,x
JP PO,x	E2,x
JP M,x	FA,x
JP P,x	F2,x

JR d Salto relativo: salta d byte (lo spostamento). (indirizzamento relativo).

JR d	18 d

JR c,d Salto relativo: salta d byte (lo spostamento) se si verifica la condizione c. c può essere Z (uguale); NZ (non uguale); C (riporto) o NC (nessun riporto). (indirizzamento relativo).

JR NZ,d	20,d
JR Z,d	28,d
JR NC,d	30,d
JR C,d	38,d

LD r,n Carica il registro r con il dato n. (indirizzamento immediato).

LD A,n	3E,n
LD B,n	06,n
LD C,n	0E,n
LD D,n	16,n
LD E,n	1E,n
LD H,n	26,n
LD L,n	2E,n

LD rr,nn Carica la coppia di registri rr con il numero di due byte nn. (ind. immediato).

LD BC,nn	01,nn
LD DE,nn	11,nn
LD HL,nn	21,nn

LD A, (x) Carica nell'accumulatore il dato posto all'indirizzo x (indirizzamento assoluto).

LD A, (x)	3A, (x)

LD rr,(x) Carica la coppia di registri rr con il contenuto degli indirizzi x e x+1. (indirizzamento assoluto).

LD BC, (x)	ED4B, (x)
LD DE, (x)	ED5B, (x)
LD HL, (x)	2A, (x)

LD A,r Carica l'accumulatore con il contenuto del registro r. (indirizzamento implicito).

LD A,A	7F
LD A,B	78
LD A,C	79
LD A,D	7A
LD A,E	7B
LD A,H	7C
LD A,L	7D

LD B,r Carica il registro B con il contenuto del registro r. (indirizzamento implicito).

LD B,A	47
LD B,B	40
LD B,C	41
LD B,D	42
LD B,E	43
LD B,H	44
LD B,L	45

LD C,r Carica il registro C con il contenuto del registro r. (indirizzamento implicito).

LD C,A	4F
LD C,B	48
LD C,C	49
LD C,D	4A
LD C,E	4B
LD C,H	4C
LD C,L	4D

LD D,r Carica il registro D con il contenuto del registro r. (indirizzamento implicito).

LD D,A	57
LD D,B	50
LD D,C	51
LD D,D	52
LD D,E	53
LD D,H	54
LD D,L	55

LD E,r Carica il registro E con il contenuto del registro r. (indirizzamento implicito).

LD E,A	5F
LD E,B	58
LD E,C	59
LD E,D	5A
LD E,E	5B
LD E,H	5C
LD E,L	5D

LD H,r Carica il registro H con il contenuto del registro r. (indirizzamento implicito).

LD H,A	67
LD H,B	60
LD H,C	61
LD H,D	62
LD H,E	63
LD H,H	64
LD H,L	65

LD L,r Carica il registro L con il contenuto del registro r. (indirizzamento implicito).

LD L,A	6F
LD L,B	68
LD L,C	69
LD L,D	6A
LD L,E	6B
LD L,H	6C
LD L,L	6D

LD r,(rr) Carica il registro r con il contenuto dell'indirizzo posto nella coppia di registri rr. (indirizzamento indiretto).

LD A, (BC)	0A
LD A,(DE)	1A
LD A,(HL)	7E
LD B,(HL)	46
LD C,(HL)	4E
LD D,(HL)	56
LD E,(HL)	5E
LD H,(HL)	66
LD L,(HL)	6E

LD(x),A Deposita il contenuto dell'accumulatore nell'indirizzo x. (indirizzamento assoluto).

LD (x),A	32,x

LD(x),rr Deposita il contenuto della coppia di registri rr agli indirizzi x e x+1.(indirizzamento assoluto)

LD (x),BC	ED43,x
LD (x),DE	ED53,x
LD (x),HL	22,x

LD(rr),r deposita il contenuto del registro r all' indirizzo nella coppia di registri rr. (indiciz. indiretto.)

LD (BC),A	02
LD (DE),A	12
LD (HL),A	77
LD (HL),B	70
LD (HL),C	71
LD (HL),D	72
LD (HL),E	73
LD (HL),H	74
LD (HL),L	75

LD(rr),n Deposita il dato n all'indirizzo posto nella coppia di registri rr. (indirizzamento immediato/indiretto).

LD (HL),n	36

RET Ritorna dalla subroutine. (indicizzamento indiretto.)

RET	C9

RET c Ritorna dalla subroutine se si verifica la condizione c. c può essere Z (uguale); NZ (non uguale); C (riporto); NC (nessun riporto); PE (parità pari); PO (parità dispari); M (negat.) o P (posit.). (ind. indiretto).

RET Z	C8
RET NZ	C0
RET C	D8
RET NC	D0
RET PE	E8
RET PO	E0
RET M	F8
RET P	F0

SBC A,n Sottrai con riporto il dato n dall'accumulatore. (indirizzamento immediato).

SBC A,n	DE,n

SBC A,r Sottrai con riporto il contenuto del registro r dall'accumulatore. (ind. implicito).

SBC A,A	9F
SBC A,B	98
SBC A,C	99
SBC A,D	9A
SBC A,E	9B
SBC A,H	9C
SBC A,L	9D

SBC HL,rr Sottrai con riporto il contenuto della coppia di registri rr dalla coppia di registri HL.(ind. implicito).

SBC HL,BC	ED42
SBC HL,DE	ED52
SBC HL,HL	ED62

SBC A,(HL) Sottrai con riporto il contenuto dell'indirizzo posto nella coppia di registri HL dall'accumulatore. (ind. indiretto).

SBC A,(HL)	9E

SCF Attiva il carry flag. (indirizzamento implicito).

SCF	37.

SUB n Sottrai il dato n dall'accumulatore. (ind. immediato).

SUB, n	D6, n

SUB r Sottrai il contenuto del registro r dall'accumulatore. (indirizzamento immediato).

SUB A	97
SUB B	90
SUB C	91
SUB D	92
SUB E	93
SUB H	94
SUB L	95

SUB(HL) Sottrai il contenuto dell'indirizzo posto in HL dall'accumulatore.(ind. indiretto).

SUB (HL)	96

Soluzioni dei problemi

Pagina 340
&A7 è in decimale 167. 513 è in esadecimale &201.

Pagina 402
1. 25+73 (25 è &19 e 73 è &49)

Consiglio: un modo semplice per ricavare il complemento a due di un numero è di sottrarlo da 256 e poi convertire il risultato in esadecimale. Per esempio: 256—6=250 che è FA in esadecimale.

Z80		**6502**		**Significato**
Mnemonici	**Cod.esad.**	**Mnemonici**	**Cod.esad.**	
LD A,&19	3E,19	LDA #&19	A9 19	Metti &19 nell'accumulat.
ADD A,&49	C6,49	ADC #&49	69 49	Somma &49 all'accumulat.
LD(indirizzo),A	32,indirizzo	STA indirizzo	8D indirizzo	Deposita il contenuto dell'accumulatore ad un certo indirizzo
RET	C9	RTS	60	Ritorna.

2.64+12+14(64 è &40, 12 è &0C e 14 è &0E)

Z80		**6502**		**Significato**
Mnemonici	**Cod.esad.**	**Mnemonici**	**Cod.esad.**	
LD A,&40	3E,40	LDA #&40	A9 40	Metti &40 nell'accumulat.
ADD A,&0C	C6,0C	ADC #&0C	69 0C	Somma &0C all'accumulat.
ADD A,&0E	C6,0E	ADC #&0E	69 0E	Somma &0E all'accumulat.
LD(indirizzo),A	32,indirizzo	STA indirizzo	8D indirizzo	Deposita il contenuto dell'accumulatore ad un certo indirizzo
RET	C9	RTS	60	Ritorna.

Termini della robotica

Androide: Tipo di robot dall'aspetto umano.

Beccheggio: Nome del movimento verticale del polso di un robot, analogo a quello di una leva.

Cellula fotoelettrica: Dispositivo elettronico che individua la luce. Vengono utilizzate spesso come componenti di sensori.

Feedback: Informazioni su un robot o su quello che lo circonda che un computer riceve dai sensori del robot.

Gradi di libertà: Termine tecnico che si riferisce alle varie direzioni in cui si può spostare un braccio di un robot. Di solito, quanti piú sono i giunti in un braccio di un robot, tanto piú sono i gradi di libertà.

Imbardata: Il nome del movimento a sinistra e destra del polso di un robot, simile a quello del manubrio di una bicicletta.

Ingranaggi: Riducono o aumentano la velocità di un motore. Sono fra un motore e la parte del robot che azionano.

Intelligenza artificiale: Lo studio della fabbricazione di macchine capaci di fare "cose intelligenti". Gli esperti sono discordi su cosa sia l'intelligenza o un comportamento intelligente.

Interfaccia: Disposta fra il robot e il suo computer per trasformare i segnali elettrici del computer in istruzioni per il robot e viceversa.

Inviluppo di lavoro: Lo spazio raggiungibile dal braccio di un robot.

LOGO: Linguaggio di programmazione usato spesso per programmare robot che disegnano, come il Turtle.

Navigazione: Modo in cui il computer utilizza le informazioni dei sensori di un robot mobile per far spostare il robot da un punto a un altro senza urtare nulla.

Odometro: Sensore che misura la distanza percorsa da un veicolo su ruote.

Pinza: Il meccanismo fissato al polso di un braccio di un robot per tenere gli oggetti.

Porta: La presa in un computer in cui vengono inserite le interfacce e altre apparecchiature elettroniche.

Programma: Una serie di istruzioni date a un computer che controllano tutto quello che fa un robot.

Programmazione lead-through: Un modo per insegnare a un robot facendogli compiere i movimenti necessari per eseguire un lavoro.

Robot: Macchina controllata da computer programmabile per eseguire operazioni di vario tipo. Gli esperti non sono d'accordo sulla definizione esatta del termine.

Rullio: Nome del movimento orizzontale del polso di un robot, analogo a quello di una barca.

Sensore: Dispositivo che fornisce al robot informazioni sul robot stesso o su quello che lo circonda.

Sensore a sonar: Spesso utilizzati nella navigazione, questi sensori emettono un suono per poi "ascoltarne" l'eco che rimbalza sugli ostacoli. Le distanze vengono calcolate in base al tempo impiegato dal suono per tornare.

Sintetizzatore del linguaggio: Dispositivo elettronico, spesso un chip, che può essere programmato per produrre parole e frasi attraverso un altoparlante. Ogni parola viene suddivisa in piccole unità sonore che vengono poi riprodotte in modo digitale.

Sistema idraulico: Dispositivo che utilizza un olio speciale contenuto in tubi e cilindri per azionare le parti meccaniche di un robot. Utilizzato spesso nei robot a braccio.

Sistema pneumatico: Dispositivo alimentato dall'aria o da un altro gas per azionare una parte meccanica di un robot, spesso la pinza.

Trasformatore: Dispositivo elettronico che trasforma l'elettricità della rete in un basso voltaggio adatto all'alimentazione di macchine come i microrobot e i trenini elettrici.

Turtle: Microrobot con ruote programmato, nel linguaggio di programmazione LOGO, per spostarsi e disegnare.

Vista meccanica: Dispositivo controllato dal computer che fornisce al robot un rudimentale senso della vista.

Modulo di gioco

Qui comincia il listato del Modulo di gioco. È molto lungo, quindi devi fare molta attenzione; fai una pausa se ti senti stanco. Accertati di includere tutte le conversioni segnate col simbolo sul tuo computer

```
2 REM *------------------*
5 REM * MODULO DI GIOCO *
6 REM *------------------*
```

```
•8 WIDTH 40:COLOR 7,0:CLS
```

WIDTH 40 permette di scegliere la risoluzione in modo testo (40x25).

```
10 GOSUB 2500
```

Richiama la routine di inizializzazione.

```
20 GOSUB 2010
```

Richiama la routine di caricamento del personaggio.

```
30 GOSUB 1770
```

Richiama la routine di caricamento del sotterraneo.

```
40 LET I$=INKEY$
```




```
50 IF I$="A" AND DX<255 THEN GOSUB 870:REM ATTACCARE
60 IF I$="C" AND F(7)>0 AND O(17)+O(18)>0 THEN GOSUB 990:REM CONGIURARE
70 IF I$="P" THEN GOSUB 1410:REM PRENDERE
80 IF I$="U" THEN GOSUB 1660:REM UNGUENTI E POZIONI DELL'EMPORIO
90 IF I$="R" THEN GOSUB 1690:REM RIVELA
100 IF I$="S" THEN GOSUB 2260:REM SALVA LO STATO DEL GIOCO
110 IF I$="B" THEN LET NF=NF-1:REM MOVIMENTO IN SENSO ANTIORARIO
120 IF I$="N" THEN LET NF=NF+1:REM MOVIMENTO IN SENSO ORARIO
130 IF NF>4 THEN LET NF=1
140 IF NF<1 THEN LET NF=4
150 IF I$="M" THEN LET NX=NX+D(NF,1):LET NY=NY+D(NF,2):REM MOVIMENTO
160 IF NY>15 THEN LET NY=15
170 IF NY<1 THEN LET NY=1
180 IF NX<1 THEN LET NX=1
190 IF NX>15 THEN LET NX=15
200 LET RH=R(NX,NY)
```

Mossa del giocatore. In dipendenza del tasto utilizzato vengono attivate le varie routine.

```
210 IF RH=C1 THEN LET X=NX:LET Y=NY:GOSUB 570:LET NX=OX:LET NY=OY:LET F(1)=F(1)-
.03
220 IF RH=C6 THEN LET TX=NX:LET TY=NY:LET TF=1
230 IF TF=1 THEN LET NX=TX:LET NY=TY
240 IF F(1)>S1*.8 AND RND(1)*8<F(6) THEN LET TF=0
250 IF I$>"" THEN LET F(1)=F(1)*.99
```

```
260 IF F(1)<S1 THEN LET F(1)=F(1)+(F(2)/1100)
270 GOSUB 480

280 IF OX<>NX OR OY<>NY THEN LET X=OX:LET Y=OY:GOSUB 570
290 LET OX=NX:LET OY=NY

300 IF DX<255 THEN GOSUB 620
310 IF F(1)>0 AND FI<1 AND RH<>C5 THEN GOTO 40

320 IF RH=C5 THEN LET M$=T$(12):GOSUB 430:LE=LE+1:GOSUB 1760:GOTO 40
```

Uscita dal livello.

```
330 IF F(1)<1 THEN GOSUB 810
```

Se non c'è forza sufficiente, il personaggio muore.

```
340 LOCATE 11,1:END
```



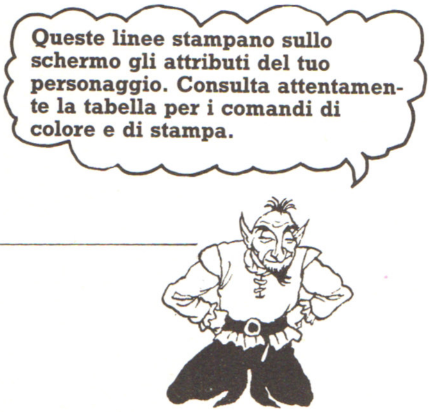

```
•350 SOUND J/2+500,.05:RETURN
•360 SOUND J/2+400,.05:RETURN
```

```
•370 COLOR 0,2
 380 LOCATE 5,1:PRINT M$
 390 LET I$=INKEY$
 400 IF I$="" THEN GOTO 390
 410 LOCATE 5,1:PRINT LEFT$(B$,W);:LET M$=""
 420 RETURN
```

Stampa il messaggio contenuto in M$ e attende che venga digitato un tasto.

```
•430 COLOR 0,2
 440 LOCATE 5,1:PRINT M$;
 450 FOR D=1 TO 200:NEXT D
 460 LOCATE 5,1:PRINT LEFT$(B$,W);:LET M$=""
 470 RETURN
```

Controllo dello schermo.




```
•480 COLOR 3,1
 490 LOCATE NY+5,NX:PRINT MID$(F$,NF,1);
•500 COLOR 0,2
 510 LOCATE 9,16:PRINT INT(F(1));" ";
 520 LOCATE 12,16:PRINT INT(F(2));" ";
 530 LOCATE 15,16:PRINT INT(F(7));" ";
 540 LOCATE 18,17:PRINT MID$("NESO.",NF,1);
 550 LOCATE 21,16:PRINT INT(F(5));
 560 RETURN
```

```
•570 COLOR 2,1

 580 LET RM=R(X,Y):LOCATE Y+5,X:PRINT CHR$(RM);
```

Stampa il carattere presente in R(X,Y).

```
 590 IF ABS(DX)<4 OR RM>C10 THEN RETURN
```

Attiva il mostro.

```
 600 LET MT=RM:LET M=MT-32:LET MV=M/16:LET MS=M*6:LET DX=3:LET LX=X:LET LY=Y
 610 RETURN
```

```
620 LET DX=LX-NX:LET SX=SGN(DX):LET DY=LY-NY:LET SY=SGN(DY)
630 LET MX=LX-(MV*SX):LET MY=LY-(MV*SY):LET RM=R(MX,MY)
640 IF RM>CO AND RM<>MT THEN LET MY=LY:LET MX=LX
650 LET R(LX,LY)=CO:LET X=LX:LET Y=LY:GOSUB 570
660 LET R(MX,MY)=MT:LET X=MX:LET Y=MY:GOSUB 570
670 LET LX=MX:LET LY=MY:LET H=0
680 IF ABS(DX)<=1 AND ABS(DY)<=1 AND RH <>C7 THEN LET H=M*.5:LET J=H:GOSUB 350
690 IF H*12<F(6)+F(3) THEN RETURN
700 LET M$=T$(5):GOSUB 430:GOSUB 360
710 LET H=H/(3+O(9)+O(10)+O(11)+O(12)+O(13)+O(14))
720 LET F(1)=F(1)-H:LET F(2)=F(2)-(H/101)
730 LET I=1:LET WB=0:LET MB=RND(1)*M
740 LET J=MT:GOSUB 350:GOSUB 360
750 IF MB=1 AND O(I)>0 THEN GOSUB 780
760 IF I<11 THEN LET I=I+1:GOTO 750
770 RETURN
780 LET O(I)=0:LET M$=T$(8)+" "+W$(I):GOSUB 430
790 LET MB=0:GOSUB 360:LET J=1:GOSUB 350
800 RETURN
```

Attacco del mostro.

```
810 LET NF=5:LET F(1)=0:GOSUB 440
820 LOCATE 5,2:PRINT "SEI MORTO !"
830 FOR J=150 TO 1 STEP -4
```

Morte del personaggio.

Queste sono subroutine per i suoni. Le linee di questo tipo contengono le istruzioni per produrre il suono.

```
840 GOSUB 350:GOSUB 360:GOSUB 570:GOSUB 480
850 NEXT J
860 RETURN
```



```
870 LET M$=T$(RND(1)*3):GOSUB 360
880 LET H=F(1)+O(1)+O(2)+O(3)+O(4)+O(5)+O(7)+O(8)+(RND(1)*F(6))
890 IF F(3)+F(6)<(RND(1)*M+2) THEN LET M$=T$(4):LET HT=0
900 LET MS=MS-H:GOSUB 430
910 LET F(1)=F(1)-(H/100):LET F(5)=F(5)+.05
920 IF MS<1 THEN GOSUB 940
930 RETURN
```

Attacco del personaggio.

```
940 LET DX=255:LET MS=0:LET R(MX,MY)=CO
950 LET F(5)=F(5)+.1
960 LET M$=T$(6):GOSUB 430
970 FOR J=200 TO 150 STEP -8:GOSUB 350:GOSUB 360:NEXT J
980 GOSUB 570:RETURN
```

Morte del mostro.

```
•990 GOSUB 480:COLOR 0,2
1000 LOCATE 2,1:PRINT "PUOI USARE LA MAGIA";
1010 IF O(17)>0 THEN LOCATE 3,1:PRINT "DEL NECRONOMICON";
1020 IF O(18)>0 THEN LOCATE 3,1:PRINT "DELLE PERGAMENE";
1030 LOCATE 4,1:PRINT "CONSULTA IL LIBRO";
1040 LET M$="INCANTESIMO NRO.?":GOSUB 370
1050 LET SL=VAL(I$)
```

Controllano se il personaggio è in possesso del Necronomicon e/o delle Pergamene.

Qui iniziano le routine per le magie.



```
1060 IF SL=0 OR (O(17)=0 AND SL<5) OR (O(18)=0 AND SL>3) OR SL>6 THEN GOTO 1040
1070 LET M(SL)=M(SL)-1:LET X=NX:LET Y=NY
1080 IF M(SL)<0 THEN LET M$=T$(9):LET SL=7
1090 FOR J=1 TO 4:LOCATE J+1,1:PRINT LEFT$(B$,W);:NEXT J:GOSUB 570
```

Controllano se l'incantesimo (SL) è valido o esaurito, e fanno diminuire di 1 il numero degli incantesimi.

```
1100 ON SL GOSUB 1140,1190,1220,1280,1300,1390,1120
```

Scelta degli incantesimi (da 1 a 6); il settimo (e successivi) non viene accettato.

```
1110 LET F(5)=F(5)+.2
1120 GOSUB 430
1130 RETURN
```

```
1140 FOR J=1 TO 12
1150 GOSUB 350:GOSUB 360
1160 NEXT J
1170 IF DX<255 THEN LET X=MX:LET Y=MY:GOSUB 940
1180 RETURN
```

Superzap (incantesimo 1): uccide il mostro.

```
1190 IF RH=C0 THEN LET R(NX,NY)=C7
1200 LET J=100:GOSUB 350:LET J=200:GOSUB 350
1210 RETURN
```

Santuario (incantesimo 2).

```
1220 LET GX=INT(RND(1)*13+1):LET GY=INT(RND(1)*13+1)
1230 FOR J=0 TO 255 STEP 8
1240 GOSUB 360:GOSUB 350
1250 NEXT J
1255 IF R(GX,GY)<>C1 THEN R(NX,NY)=C0:NX=GX:NY=GY
1260 GOSUB 480
1270 RETURN
```

Telecinesi (incantesimo 3).

```
1280 LET F(2)=F(2)+(RND(1)*M(SL)):LET F(1)=F(1)+(RND(1)*M(SL)):LET F(7)=F(7)-1
1290 RETURN
```

Sferzata d'energia (incantesimo 4).

```
1300 FOR J=1 TO 30
1310 LET R(NX,NY)=(RND(1)*7+1)+C0
1320 GOSUB 350:GOSUB 570
1330 NEXT J
1340 IF RH>C10 THEN LET DX=255:LET MS=0
1350 FOR J=1 TO 20 STEP 4
1360 GOSUB 350
1370 NEXT J
1380 RETURN
```

Metamorfosi (incantesimo 5).

```
1390 LET F(2)=S2:LET F(1)=S1:LET F(7)=F(7)-1
1400 RETURN
```

Guarigione (incantesimo 6).

```
1410 LET GX=NX+D(NF,1):LET GY=NY+D(NF,2)
```



NF indica la direzione in cui è rivolto il personaggio.

```
1420 IF GX<0 THEN LET GX=0
1430 IF GY<0 THEN LET GY=0
1440 IF GX>15 THEN LET GX=15
1450 IF GY>15 THEN LET GY=15
1460 LET GT=R(GX,GY):IF GT=C2 OR GT=C3 OR GT=C4 THEN LET R(GX,GY)=C0
1470 IF GT=C2 THEN LET O(23)=O(23)+1:LET O(24)=O(24)+1
1480 IF GT=C3 THEN LET TR=TR+1
1490 IF GT=C4 THEN GOSUB 1550
1500 LET X=GX:LET Y=GY:GOSUB 570
1510 IF GT=C1 OR GT=C3 OR GT=C4 THEN LET J=GT:GOSUB 350:LET J=GT+5:GOSUB 350
1520 RETURN
```

Routine "Prendi". Se si prende l'idolo, il controllo va alla linea 1550.

```
1530 LET J=O(24)+O(23)*10:GOSUB 350
1540 RETURN
```

Queste linee vengono usate quando l'Idolo è stato trovato nell'ultimo sotterraneo e la Ricerca è finita. La linea 1630 calcola il tuo punteggio finale sommando i quozienti dell'esperienza, forza, vitalità, agilità, e il numero di tesori. Puoi cercare di totalizzare un punteggio piú alto la prossima volta che giochi.

```
•1550 COLOR 1,2
 1560 LOCATE 2,1:PRINT "LA RICERCA E' FINITA!"
 1570 FOR I=1 TO 18
 1580 LET J=T(I):GOSUB 350
 1590 LET X=NX:LET Y=NY
 1600 FOR N=1 TO 4:LET NF=N:GOSUB 480:NEXT N
 1610 NEXT I
 1620 LET MS=0
 1630 LOCATE 3,2:PRINT "PUNTEGGIO= ";
 1635 PRINT INT((TR*10)+(GC*F(5))+F(1)+F(2)+F(3))
 1640 LET FI=1
 1650 END
```



```
1660 IF O(24)>0 AND F(1)<S1 THEN LET F(1)=S1:LET O(24)=O(24)-1
1670 IF O(23)>0 AND F(2)<S2 THEN LET F(2)=S2:LET O(23)=O(23)-1
1680 RETURN
```

Unguenti e pozioni.

```
1690 IF LT=0 THEN LET M$=T$(7):GOSUB 430:RETURN
1700 FOR Y=NY-3 TO NY+3
1710 FOR X=NX-3 TO NX+3
1720 IF (X>0 AND X<16) AND (Y>0 AND Y<16) THEN GOSUB 570
1730 NEXT X:NEXT Y
1740 LET LT=LT-1
1750 RETURN
```

Rivela o illumina.

```
1760 IF F(5)<S3+1 THEN LET M$=T$(11):LET NX=OX:LET NY=OY:GOSUB 430:RETURN
```

Caricamento del sotterraneo.



```
•1770 COLOR 7,0:CLS:LOCATE 3,1:PRINT "PREPARA NASTRO SOTTERRANEO"
 1780 LET M$=T$(10):GOSUB 370
•1785 FILE$="LIVELLO"+CHR$(LE+48)
•1790 OPEN FILE$ FOR INPUT AS #1
•1800 LINE INPUT #1,S$
•1810 CLOSE #1
 1820 LET I=1
 1830 FOR Y=1 TO 15
 1840 FOR X=1 TO 15
 1850 LET R(X,Y)=ASC(MID$(S$,I,1))
 1860 LET I=I+1
 1870 NEXT X
 1880 NEXT Y
 1890 LET IX=ASC(MID$(S$,I,1))-OS
 1900 LET IY=ASC(MID$(S$,I+1,1))-OS
 1910 LET LE=ASC(MID$(S$,I+2,1))-48
 1920 IF LE>F(5) THEN GOSUB 1960:GOTO 1760
 1930 GOSUB 2790
 1940 LET NX=IX:LET NY=IY:LET OX=NX:LET OY=NY:LET DX=255
 1950 RETURN
 1960 PRINT:PRINT "LIVELLO TROPPO PROFONDO"
 1970 PRINT "RIAVVOLGI IL NASTRO"
 1980 PRINT "ALLA POSIZIONE"
 1990 PRINT "DEL LIVELLO ";INT(F(5))
 2000 RETURN
```

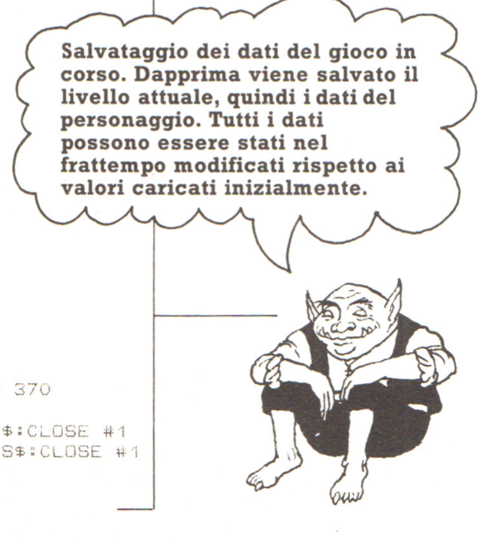



```
 2010 CLS:LOCATE 3,1:PRINT "PREPARA NASTRO EROE"
 2020 LET M$=T$(10):GOSUB 370
•2030 OPEN "EROE" FOR INPUT AS #1
•2040 LINE INPUT #1,S$
•2050 CLOSE #1
 2060 LET P=2
 2070 LET OT=ASC(MID$(S$,1,1))-AS
 2080 FOR I=1 TO 8
 2090 LET F(I)=ASC(MID$(S$,P,1))-AS
 2100 LET P=P+1
 2110 NEXT I
 2120 FOR I=1 TO OT
 2130 LET O(I)=ASC(MID$(S$,P,1))-AS
 2140 LET P=P+1
 2150 NEXT I
 2160 LET GC=ASC(MID$(S$,P,1))-AS
 2170 LET TR=ASC(MID$(S$,P+1,1))-AS
 2180 LET C$=RIGHT$(S$,LEN(S$)-(P+1))
 2190 LET S1=F(1):LET S2=F(2):LET S3=F(5)
 2200 FOR I=1 TO 2
 2210 FOR J=1 TO 3
 2220 LET M((I-1)*3+J)=O(16+I)*F(7)
 2230 NEXT J:NEXT I
 2240 IF O(16)=1 THEN LET LT=20
 2250 RETURN
```

Caricamento del personaggio.




```
 2260 LET M$="UN MOMENTO, PREGO":GOSUB 430
 2270 LET S$="":LET T$=""
 2280 FOR Y=1 TO 15
 2290 FOR X=1 TO 15
 2300 LET T$=T$+CHR$(R(X,Y))
 2310 NEXT X:NEXT Y
 2320 LET T$=T$+CHR$(NX+OS)
 2330 LET T$=T$+CHR$(NY+OS)
 2340 LET T$=T$+CHR$(LE+48)
 2350 LET S$=S$+CHR$(OT+AS)
 2360 FOR I=1 TO 8
 2370 LET S$=S$+CHR$(F(I)+AS)
 2380 NEXT I
 2390 FOR I=1 TO OT
 2400 LET S$=S$+CHR$(O(I)+AS)
 2410 NEXT I
 2420 LET S$=S$+CHR$(GC+AS)
 2430 LET S$=S$+CHR$(TR+AS)
 2440 LET S$=S$+C$
 2450 LET M$="UN TASTO PER REGISTRARE":GOSUB 370
•2455 LET FILE$="LIVELLO"+CHR$(LE+48)
•2460 OPEN FILE$ FOR OUTPUT AS #1:PRINT #1,T$:CLOSE #1
•2470 OPEN "EROE" FOR OUTPUT AS #1:PRINT #1,S$:CLOSE #1
 2480 LET FI=1
 2490 RETURN
```



```
2500 LET C$="GIOCO DI RECITAZIONE":LET B$=""
2510 LET W=40:LET OS=43:LET LE=1
2520 FOR I=1 TO W:LET B$=B$+" ":NEXT I
```




```
2530 DIM R(15,15),F(8),O(24)
2540 DIM W$(11),T$(12)
2550 DIM M(6),D(4,2),T(18)
```

```
2560 DATA "GR SPADA","SPADA","ASCIA","MAZZA","FLAGELLO"
2570 DATA "DAGA","ARMATURA","ARMATURA"
2575 DATA "ARMATURA","ELMO","ELMETTO"
2580 FOR I=1 TO 11
2590 READ W$(I)
2600 NEXT I
```

Lettura della tabella contenente l'elenco delle armi.

```
2610 DATA "UN BEL COLPO","BEL COLPO MESSERE","IL TUO SCOPO E' BUONO"
2615 DATA "MANCATO","TI HA COLPITO!!"
2620 DATA "IL MOSTRO E' MORTO","NON C'E' LUCE"
2625 DATA "TI SI SPEZZA","INCANTESIMO ESAURITO"
2630 DATA "PREMI UN TASTO","HAI BISOGNO DI ESPERIENZA"
2635 DATA "USCITA DA QUESTO LIVELLO"
2640 FOR I=1 TO 12
2650 READ T$(I)
2660 NEXT I
```

Lettura della tabella contenente i messaggi da inviare a video. La tabella è utilizzata nelle varie subroutine.

```
2670 DATA 0,-1,1,0,0,1,-1,0
2680 FOR I=1 TO 4:READ D(I,1),D(I,2):NEXT I
```

Lettura tabella utilizzata dalla routine "Prendi".

```
2690 LET FI=0:LET DX=255:LET NF=0
2700 LET TX=0:LET TY=0:LET TF=0:LET TR=0
2710 LET MX=0:LET MY=0:LET DY=12:LET F$=""
2720 LET NX=1:LET NY=1:LET RE=0:LET LT=0
```

Inizializzazione di alcune variabili.

```
2740 LET F$=CHR$(24)+CHR$(26)+CHR$(25)+CHR$(27)+CHR$(2)
```

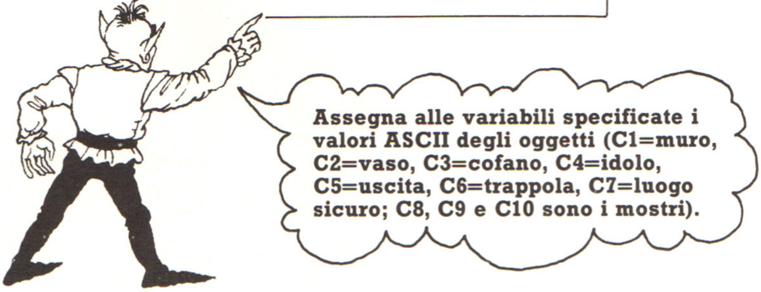
In F$ vengono registrati i codici ASCII che mostrano la direzione del personaggio. Il codice 24 rappresenta la freccia in alto, il 26 la freccia a destra, il 25 la freccia in basso, il 27 la freccia a sinistra, il 2 il personaggio morto. Essendo codici inferiori a 32, potrebbero non funzionare su altri computer. Controllate i codici ASCII sul vostro manuale, ed eventualmente sostituite quelli presenti qui con i vostri.

```
2760 DATA 69,117,73,121,81,129,69,117,73,121,81,129,89,137,97,145,101,149
2770 FOR I=1 TO 18:READ T(I):NEXT I:GOSUB 2930
2780 RETURN
```

```
•2790 COLOR 0,2:CLS
•2800 COLOR 0,3
 2810 PRINT C$;LEFT$(B$,W-LEN(C$));
•2820 COLOR 3,2
 2830 FOR I=1 TO 4:LOCATE I+1,1:PRINT LEFT$(B$,W);:NEXT I
•2840 COLOR 1,0
 2850 FOR I=1 TO 15:LOCATE 5+I,1:PRINT LEFT$(B$,15);:NEXT I
•2860 COLOR 3,1
 2870 LOCATE 8,16:PRINT "FORZA";
 2880 LOCATE 11,16:PRINT "VITAL";
 2890 LOCATE 14,16:PRINT "AURA ";
 2900 LOCATE 17,16:PRINT "DIREZ";
 2910 LOCATE 20,16:PRINT "ESPER";
 2920 RETURN
```




```
2930 REM
3170 LET AS=65:LET C0=OS+3
3180 LET C1=88:LET C2=86:LET C3=67:LET C4=73
3190 LET C5=60:LET C6=84:LET C7=76:LET C8=35:LET C9=36:LET C10=38
3200 RETURN
```





Il videodisco

Per memorizzare enormi quantità di dati in pochissimo spazio non esistono solo i Cd-Rom: sta infatti guadagnando terreno l'impiego dei videodischi, specialmente dove è importante la comunicazione per immagini.

Trovare il modo e i mezzi più appropriati per comunicare è un obiettivo che sta caratterizzando il lavoro e la vita del cittadino moderno. All'interno di questa ricerca, è nata una nuova tecnologia: quella delle memorie ottiche (così chiamate in quanto le informazioni presenti sui dischi ottici vengono incise e lette da un raggio laser), che si propongono come nuovi e efficaci mezzi di gestione dell'informazione e della comunicazione destinati prevalentemente al mercato professionale e industriale. Tra le varie memorie ottiche (Cd-Rom, Cd-Worm, Cd-Video, Cd-I, eccetera), il videodisco è l'unico che per ora permette la riproduzione, oltre che di immagini fisse, testi grafici e audio, anche di sequenze cinematografiche, rendendo questo supporto uno strumento capace di risolvere molteplici problemi nella gestione della comunicazione di informazioni rivolte sia all'interno sia all'esterno di un'azienda od organizzazione.



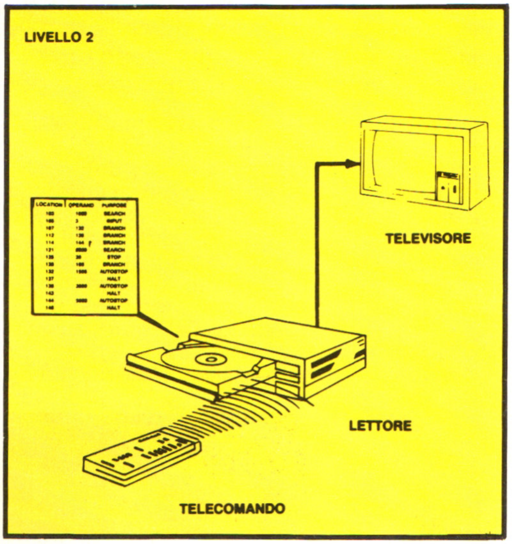
Videodisco e personal computer

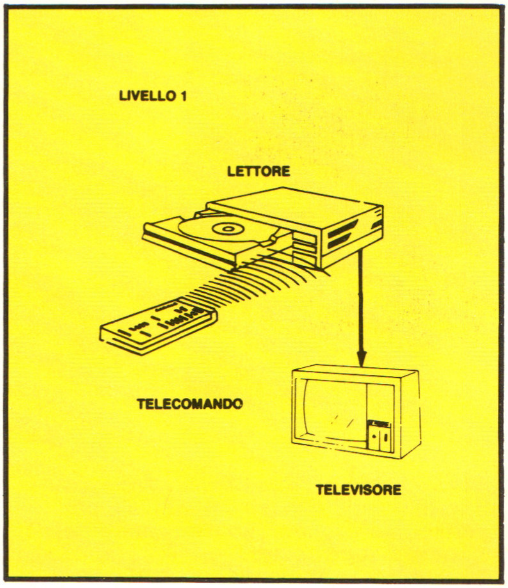
Unendo le potenzialità e versatilità di impiego dei videodischi alla capacità di gestione dei dati di un personal computer, si possono sviluppare sistemi audiovisivi interattivi, configurati ad hoc a seconda delle varie esigenze o dei campi di applicazione; per interattivi si intende un'esplorazione attiva dell'informazione, tale da permettere di intervenire su quest'ultima guidandola e orientandola secondo le proprie esigenze e capacità.
Oggigiorno esistono tre livelli di interattività dei sistemi a videodischi.
Livello uno - Livello che consente una limitata interattività a costi minimi e che si rivolge prevalentemente al mercato consumer; il sistema è formato da un semplice lettore dei dischi e da un monitor a esso collegato; l'utente, attraverso un semplice telecomando, può accedere in modo casuale a sezioni del disco che desidera vedere in meno di un secondo; le prestazioni offerte a questo livello includono scorrimento continuo delle immagini, scelta del canale audio, divisione del disco in capitoli; tale sistema presenta vantaggi di facilità d'uso, di lunga durata delle informazioni nel tempo e di bassi costi di realizzazione.
Livello due - In questo livello vengono inseriti nel videodisco dati digitali, per permettere un controllo elementare del disco con il computer.
Un tale livello di interattività permette applicazioni molto interessanti nel campo professionale, come per esempio gli archivi elettronici di banche immagini.
L'utente in questo caso ha a disposizione alcune ulteriori possibilità di ricerca, come la ricerca del numero di fotogramma, la possibilità di accedere direttamente al punto del disco desiderato, il fermo immagine, la variazione della velocità di visione delle immagini, un controllo elementare del disco con il computer.
Livello tre - In questo caso il lettore di videodischi viene comandato completamente dal computer: tale lettore lavora come una qualsiasi periferica. L'utente si trova, quindi, a interagire con un sofisticato sistema di comunicazione, di cui il videodisco è parte fondamentale; la potenza e la flessibilità di tale sistema trova limitazioni non di carattere tecnologico, ma di carattere economico, poichè il costo del sistema base (programma video, software di gestione e hardware necessario) è proporzionale al grado di sofisticazione che si vuole raggiungere.

Alto livello di interattività

Consideriamo per esempio un sistema interattivo di terzo livello: esso, normalmente, è composto da:

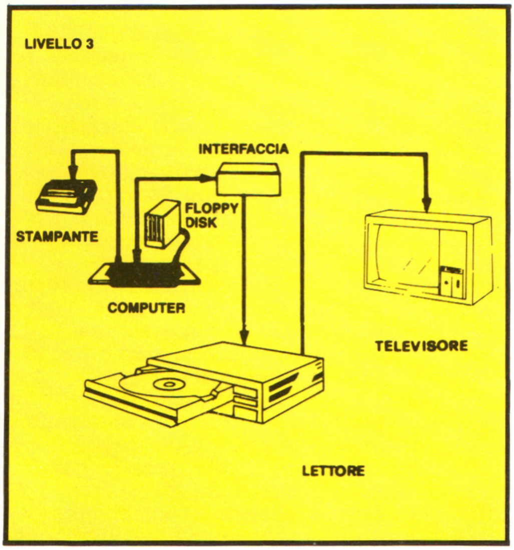
- Un videodisco (capace di contenere fino a 54 mila immagini per facciata, equivalenti a 36 minuti di filmato, più due tracce audio stereofoniche);
- un lettore di videodisco (di dimensioni simili a un videoregistratore), che sfrutta la velocità e precisione di ricerca del raggio laser;
- un personal computer;
- un monitor o televisore a colori;
- un programma software appositamente studiato e realizzato, che gestisce le sequenze filmate e/o le immagini provenienti dal videodisco.





Questo livello permette, quindi, notevolissime applicazioni nel campo della formazione professionale, della didattica, dell'archiviazione, della simulazione, della descrizione di musei, gallerie, fiere; ancora, i videodischi possono essere usati come sofisticati manuali di lavoro, di vendita, di istruzione per prodotti industriali complessi e ingombranti, piuttosto che come depliant elettronici per agenzie di viaggio o tour operator. Attualmente, quindi, i sistemi a livello tre hanno le maggiori possibilità di sviluppo e diffusione nel mercato professionale del lavoro.
Un sistema a videodisco offre infatti, rispetto ai tradizionali sistemi di memorizzazione delle informazioni, alcuni ulteriori vantaggi, quali:

- compressione di un alto numero di informazioni in uno spazio fisico ridotto;
- eliminazione di un ingente numero di documenti da uno spazio operativo;
- facile e rapido reperimento delle informazioni;
- conservazione delle informazioni per un lungo periodo di tempo (più di 50 anni);
- facile trasferimento delle informazioni residenti sul disco, su altri supporti;
- elevato grado di flessibilità nella gestione della comunicazione, grazie all'efficacia del mezzo audiovisivo.



CPC464 e 6128 fantastici computer, fantastici TV!

L. 399.000 + IVA



TUTTO COMPRESO.

CPC464GT 64 Kb RAM con monitor fosfori verdi, tastiera, registratore a cassetta, joystick, 100 programmi/giochi: L. 399.000.+ IVA

CPC464CTM 64 Kb RAM con monitor a colori, tastiera, registratore a cassette, joystick, 100 programmi/giochi: L. 699.000.+ IVA

CPC6128GT 128 Kb RAM con monitor a fosfori verdi, velocissimo disk driver da 3" doppia faccia (180 Kb + 180 Kb), joystick, 50 programmi/giochi: L. 699.000.+ IVA

CPC6128CTM 128 Kb RAM con monitor a colori, velocissimo disk driver da 3" doppia faccia (180 Kb + 180 Kb), joystick, 50 programmi/giochi: L. 899.000.+ IVA

WKS 6128 TV.

Stazione completa composta da: CPC 6128 CTM; Tavolo a ripiani; Sintonizzatore TV; Antenna amplificata.
Tutto a L. 999.000.+ IVA

PRONTO AMSTRAD.

Telefonaci: 02/26410511, avrai ogni informazione; oppure scrivici: Casella Postale 10794 - 20124 Milano.

LI TROVI QUI.

Presso i numerosissimi punti vendita Amstrad. Cerca quello più vicino su "Amstrad Magazine" in edicola, chiedi anche Junior Amstrad la rivita che ti regala i giochi per CPC (troverai molte notizie in più).
Oltre 150 Centri di Assistenza Tecnia.

FANTASTICO, DIVENTA TV COLOR.

Al momento del tuo acquisto puoi trasformare il tuo CPC con monitor a colori in TV color, il tuo TV color, come? Ma è semplice, basta Acquistare il sintonizzatore TV (MP3) a L. 199.000.+ IVA



AMSTRAD

DALLA PARTE DEL CONSUMATORE

OPINIONE Comunicazione Integrata